PATENT APPLICATION BASED ON:

Docket No: 87018DMW

Inventors: Zhaohui Sun

                                        Gabriel Fielding

                                        Majid Rabbani

Attorney: David M. Woods

METHOD AND SYSTEM FOR VIDEO FILTERING WITH JOINT MOTION AND NOISE ESTIMATION

Commissioner for Patents
Attn: Box Patent Application
P.O. Box 1450
Alexandria, VA 22313-1450

Express Mail Label No: *EV293510375US*

Date: *November 17, 2003*

METHOD AND SYSTEM FOR VIDEO FILTERING WITH JOINT MOTION AND NOISE ESTIMATION

FIELD OF THE INVENTION

5   The invention relates generally to the field of digital video and image sequence processing, and in particular to video filtering and noise reduction in a video image sequence.

BACKGROUND OF THE INVENTION

10   With the advance of digital technologies, especially the widespread use and availability of digital camcorders, digital video is getting easier and more efficient to use in a wide variety of applications, such as entertainment, education, medicine, security, and military. Accordingly, there is an increasing demand for video processing techniques, such as noise reduction.

15   There is always certain level of noise captured in a video sequence. The sources are numerous, including electronic noise, photon noise, film grain noise, and quantization noise. The noise adversely affects video representation, storage, display, and transmission. It contaminates visual quality, decreases coding efficiency (with increased entropy), increases transmission bandwidth, and

20   makes content description less discriminative and effective. Therefore, it is desirable to reduce the noise while preserving video content.

After years of effort, video filtering still remains as a challenging task. Most of the time, the only information available is the input noisy video. Neither the noise-free video nor the error characteristics are available. To

25   effectively reduce the random noise, motion estimation is necessary to enhance temporal correlation, by establishing point correspondence between video frames. However, motion estimation itself is an under-constrained and ill-posed problem, especially when there is noise involved. Perfect motion estimation is almost impossible or not practical. Meanwhile, spatiotemporal filtering is also necessary

30   to actually reduce the random noise. The filter design heavily depends on the knowledge of the noise characteristics (which are usually not available).

Furthermore, video processing requires tremendous computational power because of the amount of data involved.

Research on noise estimation and reduction in a video sequence has been going on for decades. "Noise reduction in image sequence using motion-compensated temporal filtering" by E. Dubois and M. Sabri, IEEE Trans. on Communication, 32(7):826-831, 1984, presented one of the earliest schemes using motion for noise reduction. A comprehensive review of various methods is available in "Noise reduction filters for dynamic image sequence: a review" by J.C. Brailean, et al., Proceedings of the IEEE, 83(9):1272--1292, September 1995. A robust motion estimation algorithm is presented in "The robust estimation of multiple motions: parametric and piecewise smooth flow fields" by M. Black and P. Anandan, Computer Vision and Image Understanding, 63:75-104, January 1996.

In addition, the following patent publications bear some relevance to this area; each of which are incorporated herein by reference. Commonly-assigned U.S. Published Patent Application No. 20020109788, "Method and system for motion image digital processing" by R. Morton et al., discloses a method to reduce film grain noise in digital motion signals by using a frame averaging technique. A configuration of successive motion estimation and noise removal is employed. U.S. Patent No. 6,535,254, "Method and device for noise reduction" to K. Olsson et al., discloses a method of reducing noise in a video signal. U.S. Patent No. 6,281,942, "Spatial and temporal filtering mechanism for digital motion video signals" to A. Wang, discloses a digital motion video processing mechanism of adaptive spatial filtering followed by temporal filtering of video frames. U.S. Patent No. 5,909,515, "Method for the temporal filtering of the noise in an image of a sequence of digital images, and device for carrying out the method" to S. Makram-Ebeid, discloses a method for temporal filtering of a digital image sequence. Separate motion and filtering steps were taken in a batch mode to reduce noise. U.S. Patent No. 5,764,307, "Method and apparatus for spatially adaptive filtering for video encoding" to T. Ozcelik et al., discloses a

method and an apparatus for spatially adaptive filtering a displaced frame difference and reducing the amount of information that must be encoded by a video encoder without substantially degrading the decoded video sequence. The filtering is carried out in the spatial domain on the displaced frames (the motion

5   compensated frames). The goal is to facilitate video coding, so that the compressed video has reduced noise (and smoothed video content as well). U.S. Patent No. 5,600,731, "Method for temporally adaptive filtering of frames of a noisy image sequence using motion estimation" to M.I. Sezan et al., discloses a temporally adaptive filtering method to reduce noise in an image sequence.

10   Commonly-assigned U.S. Patent No. 5,384,865, "Adaptive, hybrid median filter for temporal noise suppression" to J. Loveridge, discloses a temporal noise suppression scheme utilizing median filtering upon a time-varying sequence of images.

In addition, International Publication No. WO94/09592, "Three

15   dimensional median and recursive filtering for video image enhancement" to S. Takemoto et al., discloses methods for video image enhancement by spatiotemporal filtering with or without motion estimation. International Publication No. WO01/97509, "Noise filtering an image sequence" to W. Bruls et al., discloses a method to filter an image sequence with the use of estimated noise

20   characteristics. Published European Patent Application EP0840514, "Method and apparatus for prefiltering of video images" to M. Van Ackere et al., discloses a method for generating an updated video stream with reduced noise for video encoding applications. European Patent Specification EP0614312, "Noise reduction system using multi-frame motion estimation, outlier rejection and

25   trajectory correction" to S.-L. Iu, discloses a noise reduction system.

One of the common features of the previously disclosed schemes is the use of independent and separate steps of motion estimation and spatiotemporal filtering. Motion estimation is taken as a preprocessing step in a separate module before filtering, and there is no interaction between the two modules. If the

30   motion estimation fails, filtering is carried out on a collection of uncorrelated

samples, and there is no way to recover from such a failure. Also there is no attempt to explicitly estimate the noise levels, leading to a high chance of mismatch between the noise in the video and the algorithms and the parameters used for noise reduction. Furthermore, a robust method has not been used in video

5 filtering, and the performance suffers when the underlying model and assumptions are violated occasionally, which happens when that data is corrupted by noise.

SUMMARY OF THE INVENTION

It is an objective of this invention to provide a robust video filtering

10 method to reduce random noise in a video sequence.

It is another objective of this invention to make the computational method robust to occasional model violations and outliers.

It is yet another objective of this invention to successively improve the performance of motion estimation, spatiotemporal filtering and noise

15 estimation through iterations.

The present invention is directed to overcoming one or more of the problems set forth above. Briefly summarized, according to one aspect of the present invention, the invention resides in a method for video filtering of an input video sequence by utilizing joint motion and noise estimation, where the filtering

20 is based on determining the noise level, as characterized by the standard deviation, of the input video sequence as corrupted by unknown noise. The method comprises the steps of:(a) generating a motion-compensated video sequence from the input video sequence and a plurality of estimated motion fields; (b) spatiotemporally filtering the motion compensated video sequence, thereby

25 producing a filtered, motion-compensated video sequence; (c) estimating a standard deviation from the difference between the input video sequence and the filtered, motion-compensated video sequence, thereby producing an estimated standard deviation; (d) estimating a scale factor from the difference between the input video sequence and the motion compensated video sequence; and (e)

30 iterating through steps (a) to (d) using the scale factor previously obtained from

step (d) to generate the motion-compensated video sequence in step (a) and using the estimated standard deviation previously obtained from step (c) to perform the filtering in step (b) until the value of the noise level approaches the unknown noise of the input video sequence, whereby the noise level is then characterized by a

5   finally determined scale factor and standard deviation.

The advantages of the invention include: (a) automatically reducing the random noise in a video sequence without the availability of noise-free reference video and without knowledge of the noise characteristics; (b) using joint motion and noise estimation to improve filtering performance through iterations in

10   a closed loop; and (c) employing a robust method to alleviate the sensitivity of occasional model violation and outliers, in motion estimation, filter design and noise estimation.

These and other aspects, objects, features and advantages of the present invention will be more clearly understood and appreciated from a review

15   of the following detailed description of the preferred embodiments and appended claims, and by reference to the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 generally illustrates features of a system in accordance with

20   the present invention.

FIG. 2 shows a system diagram of video filtering with joint motion and noise estimation.

FIG. 3 shows the successive concatenation of motion estimation and spatiotemporal filtering.

25   FIG. 4 shows a procedure for video filtering with joint motion and noise estimation.

FIGS. 5A and 5B show respective plots of (a) a Lorentzian robust function, and (b) a Geman-McClure robust function at $\sigma = 0.1$.

FIG. 6 shows a plot of an adaptive averaging filter.

30

DETAILED DESCRIPTION OF THE INVENTION

In the following description, a preferred embodiment of the present invention will be described in terms that would ordinarily be implemented as a software program. Those skilled in the art will readily recognize that the

5   equivalent of such software may also be constructed in hardware. Because image manipulation algorithms and systems are well known, the present description will be directed in particular to algorithms and systems forming part of, or cooperating more directly with, the system and method in accordance with the present invention. Other aspects of such algorithms and systems, and hardware and/or

10   software for producing and otherwise processing the image signals involved therewith, not specifically shown or described herein, may be selected from such systems, algorithms, components and elements known in the art. Given the system as described according to the invention in the following materials, software not specifically shown, suggested or described herein that is useful for

15   implementation of the invention is conventional and within the ordinary skill in such arts.

Still further, as used herein, the computer program may be stored in a computer readable storage medium, which may comprise, for example; magnetic storage media such as a magnetic disk (such as a hard drive or a floppy disk) or

20   magnetic tape; optical storage media such as an optical disc, optical tape, or machine readable bar code; solid state electronic storage devices such as random access memory (RAM), or read only memory (ROM); or any other physical device or medium employed to store a computer program.

Before describing the present invention, it facilitates understanding

25   to note that the present invention is preferably utilized on any well-known computer system, such as a personal computer. For instance, referring to Fig. 1, there is illustrated a computer system 110 for implementing the present invention. Although the computer system 110 is shown for the purpose of illustrating a preferred embodiment, the present invention is not limited to the computer system

30   110 shown, but may be used on any electronic processing system such as found in

home computers, kiosks, retail or wholesale photofinishing, or any other system for the processing of digital images. The computer system 110 includes a microprocessor-based unit 112 for receiving and processing software programs and for performing other processing functions. A display 114 is electrically

5   connected to the microprocessor-based unit 112 for displaying user-related information associated with the software, e.g., by means of a graphical user interface. A keyboard 116 is also connected to the microprocessor-based unit 112 for permitting a user to input information to the software. As an alternative to using the keyboard 116 for input, a mouse 118 may be used for moving a selector

10   120 on the display 114 and for selecting an item on which the selector 120 overlays, as is well known in the art.

A compact disk-read only memory (CD-ROM) 124, which typically includes software programs, is inserted into the microprocessor-based unit for providing a means of inputting the software programs and other

15   information to the microprocessor-based unit 112. In addition, a floppy disk 126 may also include a software program, and is inserted into the microprocessor-based unit 112 for inputting the software program. The compact disk-read only memory (CD-ROM) 124 or the floppy disk 126 may alternatively be inserted into externally located disk drive unit 122 which is connected to the microprocessor-

20   based unit 112. Still further, the microprocessor-based unit 112 may be programmed, as is well known in the art, for storing the software program internally. The microprocessor-based unit 112 may also have a network connection 127, such as a telephone line, to an external network, such as a local area network or the Internet. A printer 128 may also be connected to the

25   microprocessor-based unit 112 for printing a hardcopy of the output from the computer system 110.

Images and videos may also be displayed on the display 114 via a personal computer card (PC card) 130, such as, as it was formerly known, a PCMCIA card (based on the specifications of the Personal Computer Memory

30   Card International Association) which contains digitized images electronically

embodied in the card 130. The PC card 130 is ultimately inserted into the microprocessor-based unit 112 for permitting visual display of the image on the display 114. Alternatively, the PC card 130 can be inserted into an externally located PC card reader 132 connected to the microprocessor-based unit 112.

5   Images may also be input via the compact disk 124, the floppy disk 126, or the network connection 127. Any images and videos stored in the PC card 130, the floppy disk 126 or the compact disk 124, or input through the network connection 127, may have been obtained from a variety of sources, such as a digital image or video capture device 134 (e.g., a digital camera) or a scanner (not shown). Images

10   or video sequences may also be input directly from a digital image or video capture device 134 via a camera or camcorder docking port 136 connected to the microprocessor-based unit 112 or directly from the digital camera 134 via a cable connection 138 to the microprocessor-based unit 112 or via a wireless connection 140 to the microprocessor-based unit 112.

15   Referring now to Fig. 2, the system diagram of video filtering with joint motion and noise estimation is illustrated. A digital video sequence V={ I(i,j,k), i=1…M, j=1…N, k=1…K} is a temporally varying 2-D spatial signal I on frame k, sampled and quantized at spatial location (i,j). The observed input video sequence $\tilde{V}$ 210 is corrupted by additive random noise $\tilde{V} = V + \mathcal{E}$ with

20   ε following a Gaussian distribution $\mathcal{N}(0, \sigma_n)$. The output is a spatiotemporally filtered video $\bar{V}$ 220 with reduced noise, which is close to the noise-free video V. As the ground truth V is not available, three closely tied operations, that is, motion estimation 260, noise estimation 250 and spatiotemporal filtering 240, are carried out iteratively in a closed loop to successively improve video filtering

25   performance. The motion estimation module 260 finds point trajectories across video frames, and therefore enhances temporal correlation. It takes the filtered video frames $\bar{V}$ as input and generates a plurality of dense motion fields between temporally adjacent frames $U_{ij}(k, k + r)$ with r = -R,…,R. The module also generates the motion compensated video $\hat{v}$ 230 from the noisy input video $\bar{V}$ and

the plurality of estimated motion fields. The motion compensated video $\hat{V}$ is spatiotemporally filtered in the module 240 by adaptive weighted averaging to reduce the random noise, yielding the filtered video 220. Robust methods are employed in both motion estimation and spatiotemporal filtering, which use the noise characteristics for scale factor selection and filter design. To this end, an explicit noise estimation module 250 is introduced to decide the noise level in the input noisy video $\tilde{V}$ and the scale factors used for robust motion estimation.

As mentioned above, the observed input video sequence $\bar{V}$ 210 is corrupted by additive random noise $\bar{V} = V + \mathcal{E}$ with ε following a Gaussian distribution $\mathcal{N}(0, \sigma_n)$. Given the additive degradation model

$$\tilde{I}(i,j,k) = I(i,j,k) + \varepsilon(i,j,k)$$

with $\varepsilon(i,j,k)$ as the independent noise term, the noise level 270, measured by the standard deviation, can be estimated from the noisy input video sequence $\tilde{V}$ and the noise-free video V, as follows:

$$\sigma_n^2 = \frac{1}{KMN} \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{n=1}^{N} (\tilde{I}(i,j,k) - I(i,j,k))^2.$$

As the ground truth V is not available, we estimate the noise level σ_n 270 from the difference between the observed input video sequence $\tilde{V}$ and the filtered video sequence $\bar{V}$ 220. The spatiotemporal filtering module 240 reduces the random noise in the motion compensated video $\hat{V}$ 230 and generates the filtered video $\bar{V}$.

Noise estimation module 250 takes both $\tilde{V}$ and $\bar{V}$ as input and estimates the noise level, as characterized by the standard deviation σ_n 270. The process is iterated in a closed-loop fashion as shown in Fig. 2, which is necessary because σ_n estimated from $\tilde{V} - \bar{V}$ is in fact the noise reduction in one pass. The iterations successively improve the spatiotemporal filtering 240 and the noise estimation 250. As temporal correlation gets stronger from improved motion fields, it leads

to better noise reduction in $\bar{V}$. As $\bar{V}$ gets closer to V, it in turn increases the accuracy of the noise and motion estimation.

Noise estimation module 250 also takes both $\bar{V}$ and $\hat{V}$ as input and estimates the scale factor, as characterized by the scale factor σ_d 280.

5 (Generally speaking, as noise in $\bar{V}$ increases, the scale factor function assigns bigger weights to more samples.) The process is iterated in a closed-loop fashion as shown in Fig. 2, which is necessary because σ_d estimated from $\bar{V} - \hat{V}$ is in fact the scale factor in one pass. The iterations successively improve the motion estimation 260 and the noise estimation 250. As temporal correlation gets

10 stronger from improved motion fields, it leads to a better scale factor in terms of $\hat{V}$. As $\hat{V}$ gets closer to V, it in turn increases the accuracy of the motion estimation. Consequently, and as shown in Figure 2, the joint motion and noise estimation process is iterated in a closed-loop fashion to successively reduce the random noise and improve video filtering performance.

15 The disclosed video filtering scheme is different from the previous video noise reduction schemes (shown in Fig. 3), which are based on the successive concatenation of motion estimation and spatiotemporal filtering. The previously disclosed schemes take two independent and separate steps, where motion estimation is taken as a preprocessing step, and there is no interaction

20 between the two modules. Numerous motion estimation algorithms can be used, such as gradient-based, region-based, energy-based, and transform-based approaches. There are also a number of filters available, including Wiener filter, Sigma filter, median filter, and adaptive weighted average (AWA) filter. In fact, both motion estimation and spatiotemporal filtering are closely tied to the noise

25 characteristics shown in Fig. 2. Joint motion and noise estimation potentially can improve the video filtering performance. Furthermore, a robust method becomes essential when noise is involved.

The disclosed video filtering scheme can be summarized in a flow chart as presented in Fig. 4. The input is the noisy video sequence 210 corrupted

by unknown noise, and the output is the filtered video sequence 220 with reduced noise, which is close to the noise-free video $\mathbf{V}$. The first step 310 initializes the filtered video $\bar{\mathbf{V}}$, the standard deviation σ_n of the noise in $\tilde{\mathbf{V}}$, and the scale factor σ_d. At a high signal to noise ratio (SNR), i.e. the noise level is relatively

5   small compared to the signal, the filtered video is initialized as the input video $\bar{\mathbf{V}}$. At a low signal to noise ratio (SNR), i.e. the image quality is poor, $\bar{\mathbf{V}}$ is initialized as the spatially filtered input video. The noise level in $\tilde{\mathbf{V}}$ is used for filter design in the spatiotemporal filtering, and the scale factor σ_d is used for robust motion estimation. After initialization, motion fields between temporally adjacent frames

10   are computed in step 320 from the filtered video, and the recovered motion is used to generate the motion compensated video $\hat{\mathbf{V}}$ from the noisy video $\tilde{\mathbf{V}}$ in step 330. Spatiotemporal filtering is carried out in step 340 to reduce the random noise, by adaptive weighted averaging in step 350. The noise level used in spatiotemporal filtering and the scale factor used in motion estimation are estimated in step 360.

15   The filtered video, noise level and scale factor are updated in step 370 for the next iteration, until the termination condition in step 380 is satisfied. The termination condition is that the change in $\bar{\mathbf{V}}$ is small enough, i.e., smaller than some predetermined threshold, or that a predetermined number of iterations has been reached. More details of the individual modules of 260, 250 and 240 will be

20 - disclosed in the following.

Referring to the motion estimation module 260 in Fig. 2, motion estimation computes the motion fields between video frames. Let

$$\mathbf{U}_{ij}(k, k+r) = (u_{ij}(k, k+r), v_{ij}(k, k+r))$$, i=1...M, j=1...N, k=1...K, r=-R...R

denote the motion vector from pixel (i,j) in frame k to its correspondence in frame

25   k+r. For each frame k, 2R temporally adjacent frames are also involved, requiring a total of RK dense motion fields (both forward and backward). The requirement can be reduced to 2K temporally adjacent $\mathbf{U}_{ij}(k, k\pm 1)$, as the others can be

computed following the chain rule, $U_{ij}(k, k+r) = \sum_{l=k}^{k+r-1} U_{ij}(l, l+1)$ if r>0,

or $U_{ij}(k, k+r) = \sum_{l=k}^{k+r+1} U_{ij}(l, l-1)$ if r<0. As motion vectors are imperfect, the

chain rule could accumulate motion errors and break the temporal correlation

needed for the following filtering.

5   The recovery of motion vectors (u,v) from a pair of images solely

based on image intensity I(i,j) is under-constrained and ill-posed. Even worse, the

observed image frames are corrupted by unknown noise. A perfect motion and

noise model is almost impossible or not practical. Therefore, a robust method

plays an essential role to reduce the sensitivity of the violations of the underlying

10   assumptions.

           We use the robust motion estimation method by Black and

Anandan to recover the motion field, which is done by minimizing the energy

function

$$
\begin{aligned}
E(u,v) \; = \; & \sum_{i=1}^{M}\sum_{j=1}^{N}\{\lambda_d\rho_d(u_{ij}\frac{\partial I_{ij}}{\partial x} + v_{ij}\frac{\partial I_{ij}}{\partial y} + \frac{\partial I_{ij}}{\partial t}, \sigma_d) \\
& + \; \lambda_s \sum_{p,q\in S} \rho_s(u_{ij} - u_{i+p,j+q}, \sigma_s) \\
& + \; \lambda_s \sum_{p,q\in S} \rho_s(v_{ij} - v_{i+p,j+q}, \sigma_s)\},
\end{aligned}
$$

15   where ρ_d and ρ_s are robust functions with scale parameters ε_d and σ_s, u and v

are the horizontal and vertical motion components, and S is the 4-neighbor or 8-

neighbor spatial support of pixel (i,j). The first term in the above equation

enforces the constant brightness constraint, i.e., the points have the same

appearance in both frames. The constraint can be approximated by optical flow

20   equation $I_x u + I_y v + I_t = 0$ following Taylor expansion. The second term

enforces the smoothness constraint such that the motion vectors vary smoothly.

Coefficients λ_d and λ_s control the relative weights of the two constraints.

           In a real dataset, especially corrupted by noise, the constraints may

not be strictly satisfied at every point, due to scene changes, illumination changes,

25   occlusions, and shadows. The occasional violations of the constant brightness and

smoothness constraints can be alleviated by using a robust method and outlier

rejection. Two robust functions for M-estimate are the Lorentzian function

$\rho_L(x, \sigma) = \log(1 + \frac{1}{2}(\frac{x}{\sigma})^2)$ and the Geman-McClure function $\rho_G(x, \sigma) = \frac{x^2}{x^2 + \sigma}$, as

shown in Fig. 6 (a) and (b). Unlike the linear and quadratic functions, robust

functions assign smaller weights to the outliers. As x increases, the influence of

an outlier tapers off. Of course, the choice of the scale factor σ has direct impact

on the performance, which decides the transition from inliers to the outliers, which

will be decided by the noise estimation module 250.

As the noise-free video is not available, we use the filtered video

$\bar{V}$, instead of the observed noisy video $\tilde{V}$, for motion estimation. Compared to

$\tilde{V}$, $\bar{V}$ has reduced noise and smoother intensity surface, which helps the

computation of gradients $\bar{I}_x$, $\bar{I}_y$, and $\bar{I}_t$, yielding smoother and more consistent

motion fields.

Referring to the spatiotemporal filtering module 240 in Fig. 2, the

noisy video is filtered by adaptive weighted averaging to reduce the random noise,

independent of the video structure. Given the recovered motion fields

$U_{ij}(k, k + r)$, with r=-R...R, the adjacent noisy frames are backward compensated

to frame k

$$\dot{I}_r(i, j, k) \Leftarrow \tilde{I}(i + u_{ij}(k, k + r), j + v_{ij}(k, k + r), k + r)$$

Bilinear interpolation is carried out on the integer grid, which has a low-pass

filtering effect.

The 2R+1 frames are then filtered by adaptive weighted average

$$\bar{I}(i, j, k) = \frac{1}{z(i, j, k)} \sum_{(p,q,r) \in S} w_{ijk}(p, q, r) \dot{I}_r(i + p, j + q, k)$$

where $z(i, j, k) = \sum_{(p,q,r) \in S} w_{ijk}(p, q, r)$ is a normalization factor, and S defines

a 3-D spatiotemporal neighborhood. As $\hat{v}$ has enhanced temporal correlation, the

weighted average can reduce the random noise, which is independent of the signal.

The filter is designed as

$$w_{ijk}(p, q, r) = 1 - \rho_G(\dot{I}_{pqr}(i, j, k) - \bar{I}(i, j, k), \tau).$$

where $\rho_G(x,\sigma) = \dfrac{x^2}{x^2 + \sigma}$ is the Geman-McClure robust function shown in Fig. 5(b). The filter, as shown in Fig. 6, has a bell shape and tapers off as the intensity discrepancy increases. Other filters are also available for the filtering purpose, such as the Wiener filter, Sigma filter, median filter, and adaptive weighted

5   average (AWA) filter.

Two parameters are involved in the filter design, namely, the spatiotemporal filtering support S and the scale factor $\mathcal{T}$. The support S is usually chosen as 1x1 or 3x3 spatial neighborhood, and 7 or 9 temporally adjacent frames (with R=3,4). As the size of S increases, it helps reduce noise, but tends to blur

10 the images at the same time. So a balance is needed, especially when the motion is not perfect. The scale factor $\mathcal{T}$ is chosen as $\tau = \sigma_n \sqrt{\sigma_n}$, where σ_n is the noise level estimated from module 250. As noise in $\tilde{V}$ increases, the robust function assigns bigger weights to more samples.

Referring to noise estimation 250 in Fig. 2, the noise level σ_n is

15 estimated from the difference between the observed video $\tilde{V}$ and the filtered video $\bar{V}$. Similarly a robust method is used to estimate the scale factor σ_d. The scale factor σ_d (k,k+r) used to estimate the motion $U_{ij}(k, k+r)$ from frames k to k+r is computed from the backward motion compensated residue on frame k using the filtered frames. Given noise-free frames and correct motion vectors, the

20 residue should be 0. Otherwise, the residue is mainly due to the random noise and occasional false motion vectors. Let us define the residue as

$$\varepsilon_d(k, k+r) = \{\bar{I}(i + u_{ij}(k, k+r), j + v_{ij}(k, k+r), k+r) -$$

$$\bar{I}(i, j, k) \mid i = 1\ldots M, j = 1\ldots N, k = 1\ldots K, r = -R\ldots R\}.$$

A robust estimate of the scale factor is available as

$$\sigma_d(k, k+r) = 1.4826 \ \mathrm{median}\{|\varepsilon_d(k, k+r) - \mathrm{median}\{\varepsilon_d(k, k+r)\}|\}$$

25

The robust video filtering scheme has been tested on video sequences degraded to various noise levels, and significant performance improvement has been achieved. A few factors have contributed to the performance improvement: (a) a robust method is employed in both motion

5   estimation and spatiotemporal filtering to accommodate occasional model violations; (b) a joint motion and noise estimation process is iterated in a closed loop for the best possible performance; and (c) explicit noise estimation is carried out for temporal correlation enhancement and noise reduction.

The method disclosed according to the invention may have a

10   number of distinct applications. For example, the video filtering may used to improve video coding and compression efficiency, due to the reduced entropy. The video filtering may also be used to minimize the storage space for a video clip or to minimize the transmission bandwidth of a video sequence. Furthermore, the video filtering may used to enhance the video presentation quality, in print or in

15   display. Additionally, the video filtering may be used to extract more distinctive and unique descriptions for efficient video management, organization and indexing. In each case, the usage of the aforementioned robust filter designs further enhances the values of these applications.

The invention has been described in detail with particular reference

20   to a presently preferred embodiment, but it will be understood that variations and modifications can be effected within the spirit and scope of the invention.

PARTS LIST

110	System
112	Microprocessor-based Unit
114	Display
116	Keyboard
118	Mouse; input device
120	Selector on display
122	Disc drive unit
124	Compact disc-read only memory
125	External network
126	Floppy disk
128	Printer
130	PC card
132	PC card reader
134	Digital image or video capture device
136	Docking port
138	Cable connection
140	Wireless connection
210	Input video sequence
220	Filtered video sequence
230	Motion compensated video
240	Spatiotemporal filtering module
250	Noise estimation module
260	Motion estimation module
270	Estimated standard deviation
280	Estimated scale factor
310	Initialization step
320	Motion field computation step
330	Motion compensated frames computation step
340	Spatiotemporal filtering step
350	Noise reduction step

360   Noise level and scale factor estimation step

370   Update step

380   termination condition checking step